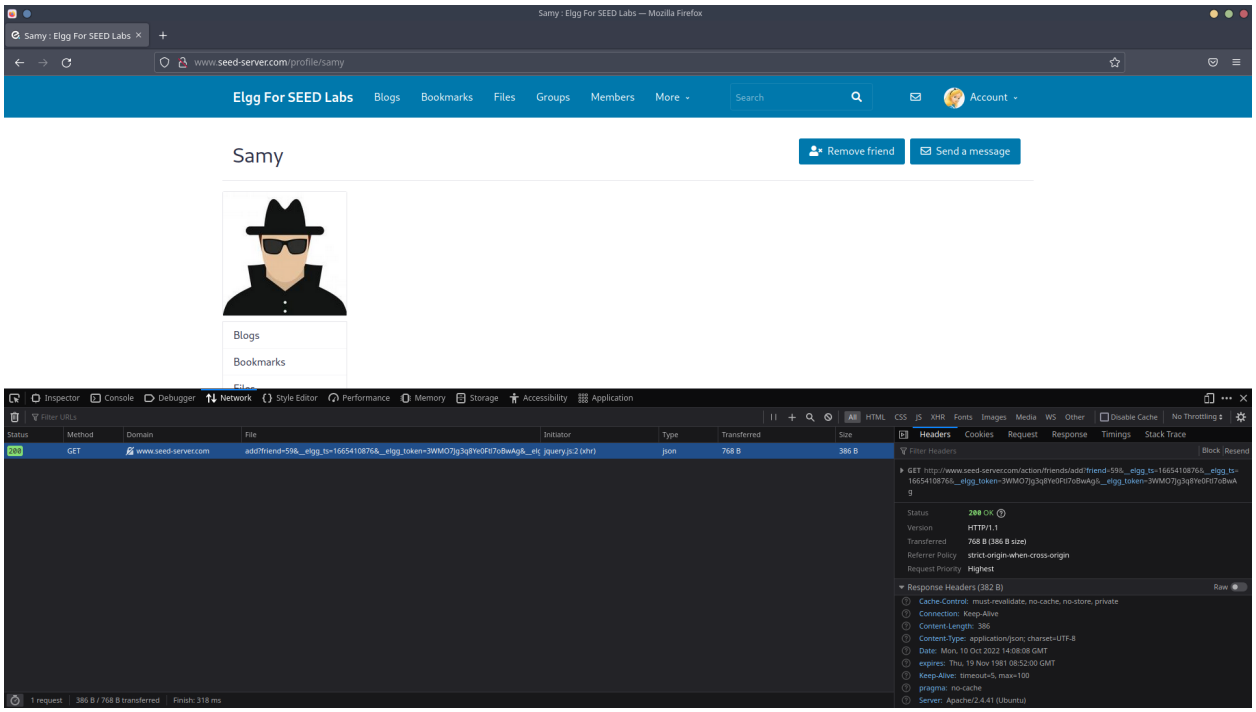
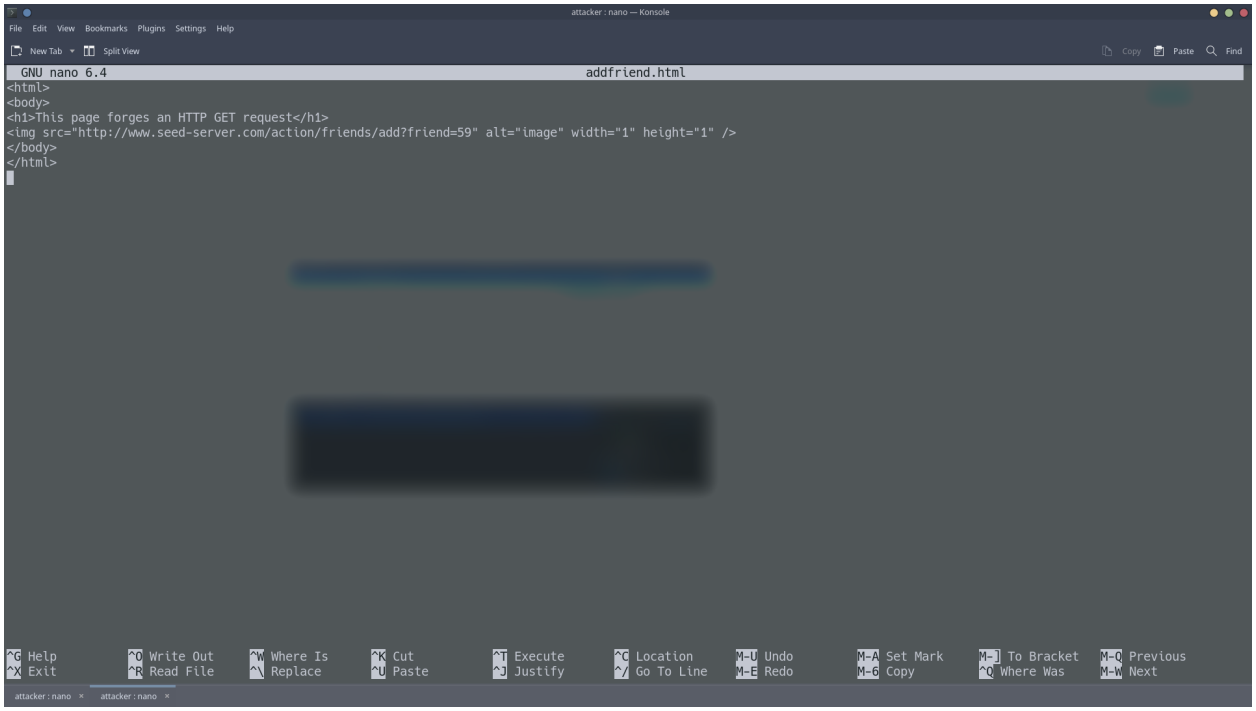
Task-1

We check the GET request which is sent to the server when Samy is added as a friend. From that request we get the URL and the guid number of Samy.

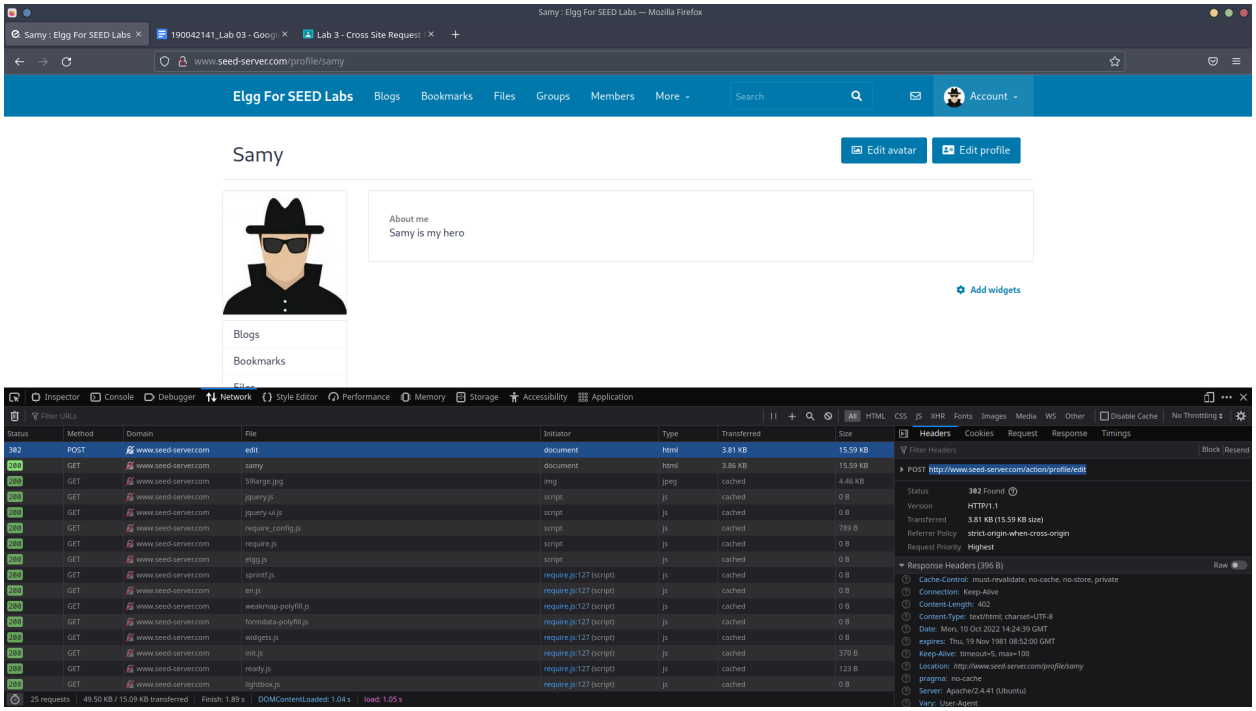


So we make a forged request using a html file where we send the request to add Samy as a friend whenever the victim visits that malicious html page.

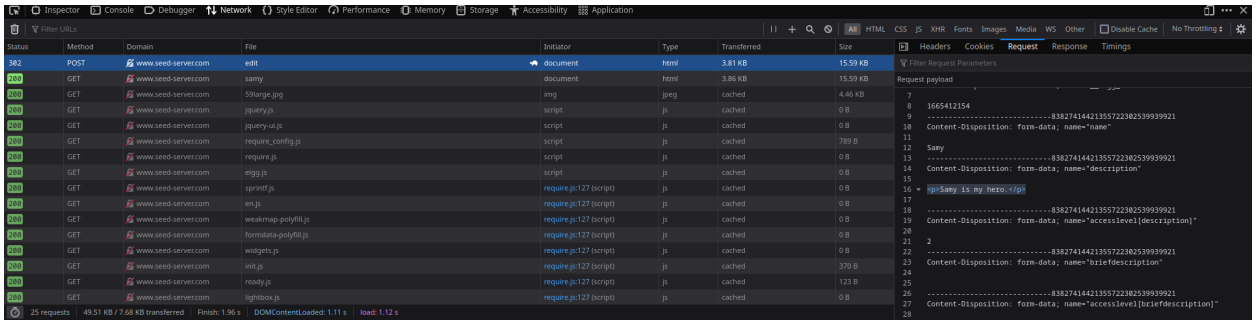


Task-2

At first we find out the URL which is required to POST any request to the profile using Samy's profile. We get the required URL for editing any profile- **`http://www.seed-server.com/action/profile/edit`**

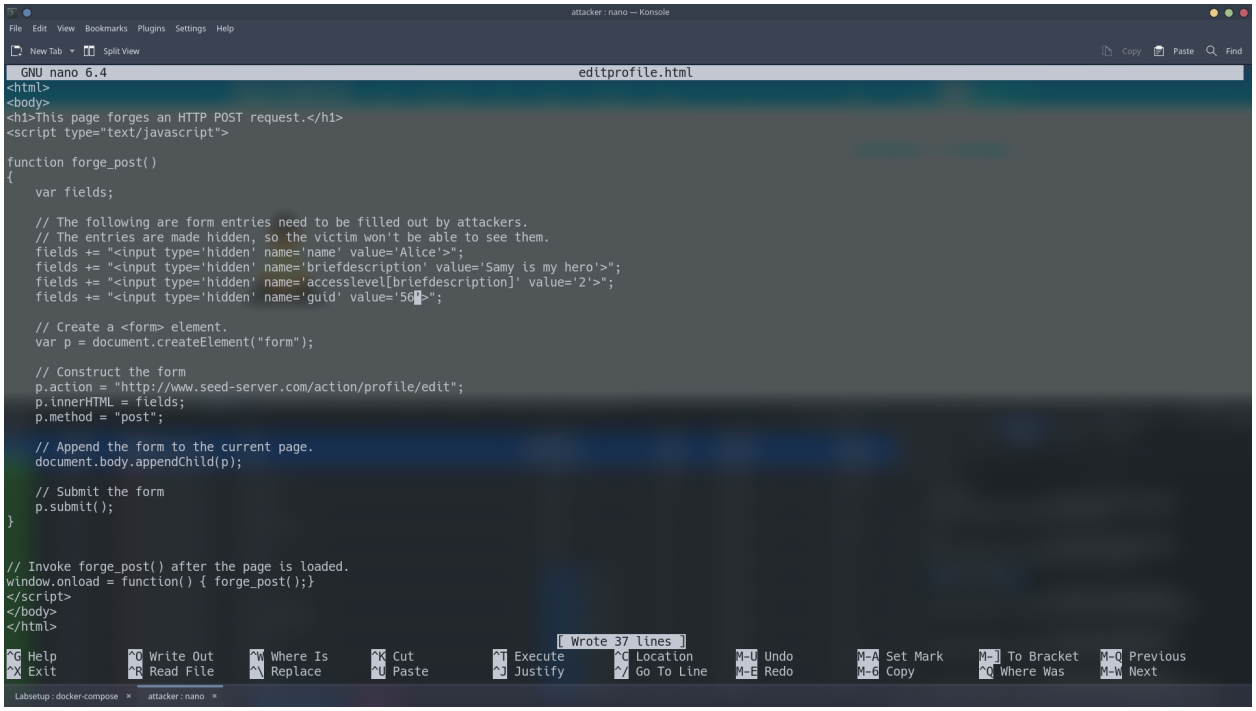


Later we check which section in the POST method "Samy is my hero" is being added

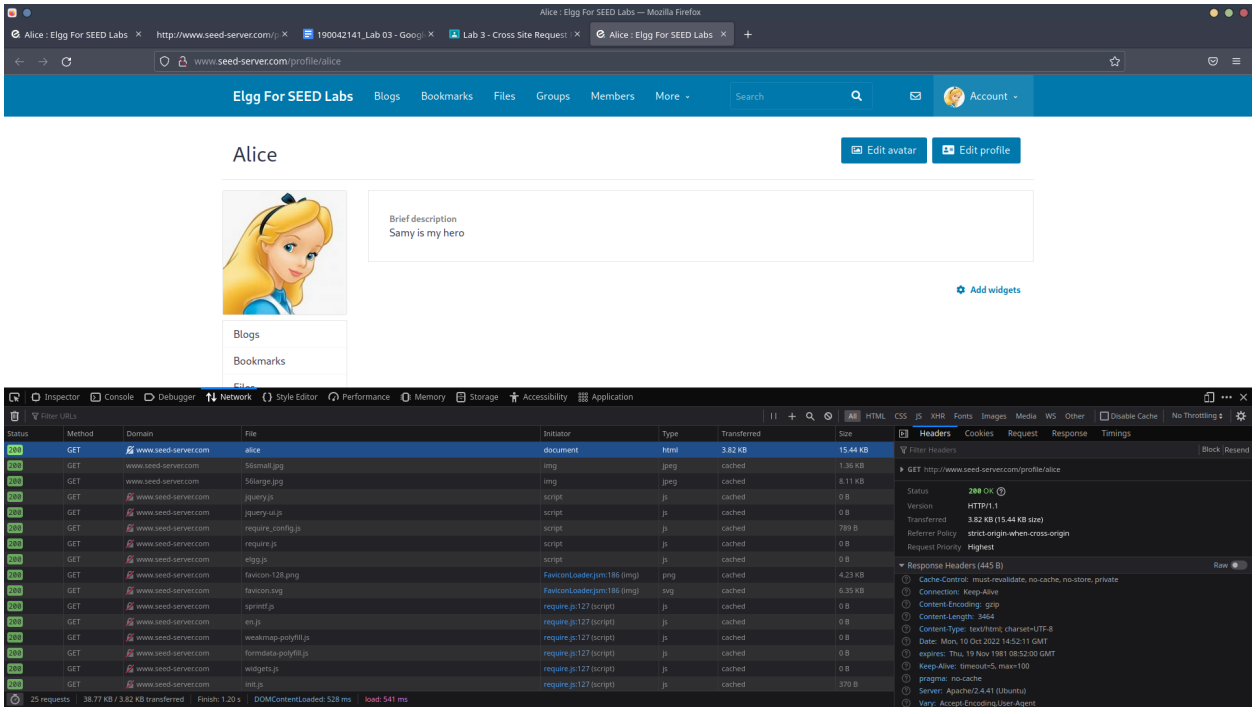


Later we find the guid number of Alice from page source

As we now know the required fields where we need to edit, we create a malicious html page in which when Alice clicks her profile will be edited by making a forged request

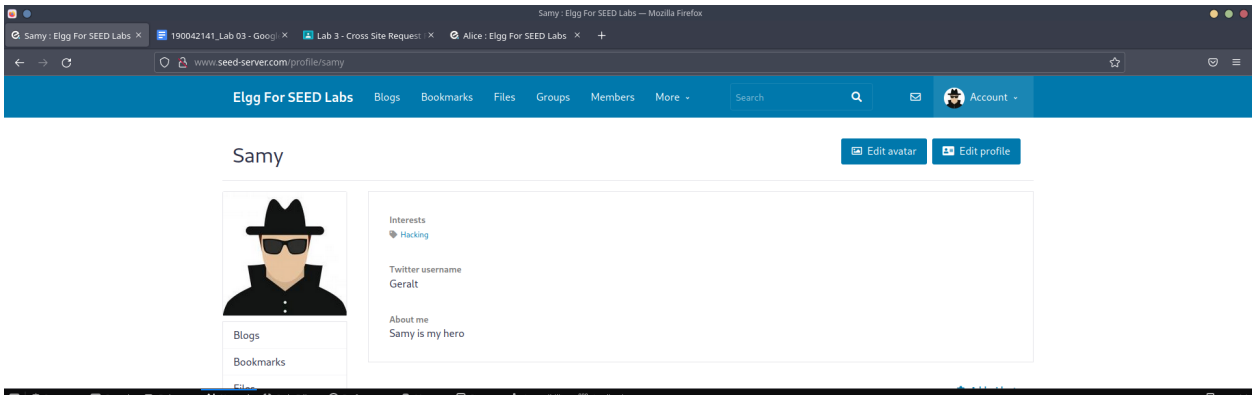


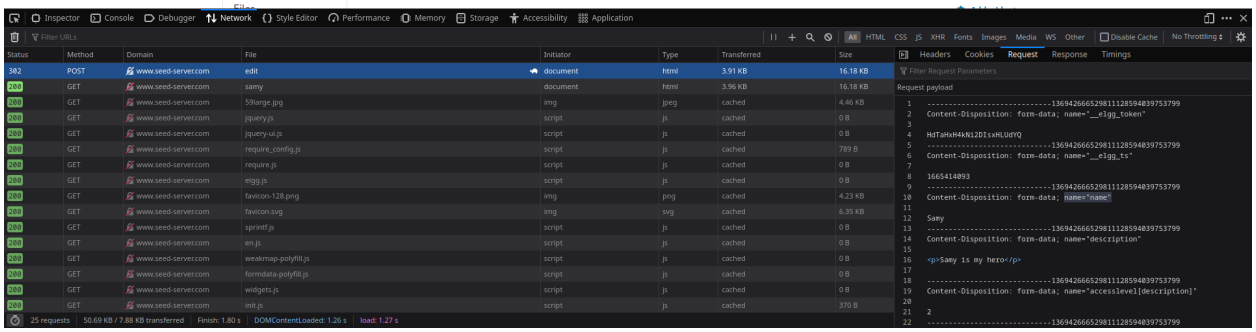
The updated profile of Alice after clicking the malicious html page

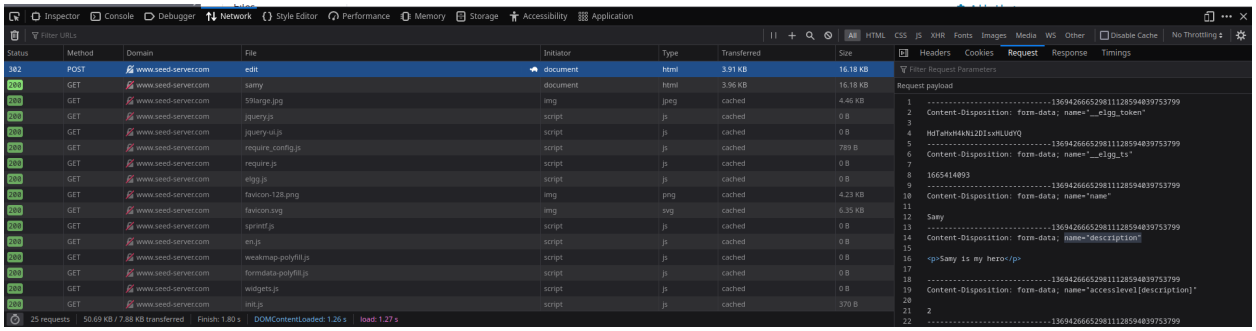


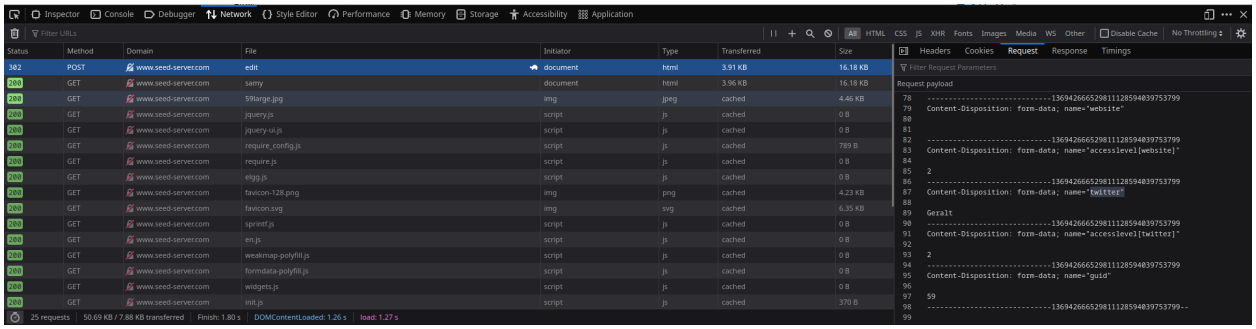
Task-3

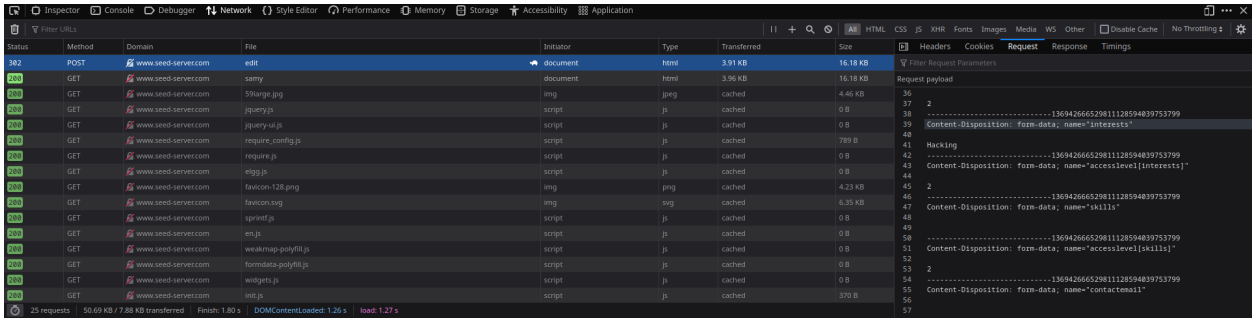
Same as the previous task we get the profile edit URL link. When we edit the profile of Samy in the fields of brief description, interests, twitter according to the requirements of the task. Later we check the fields in the html code where modifications are made.



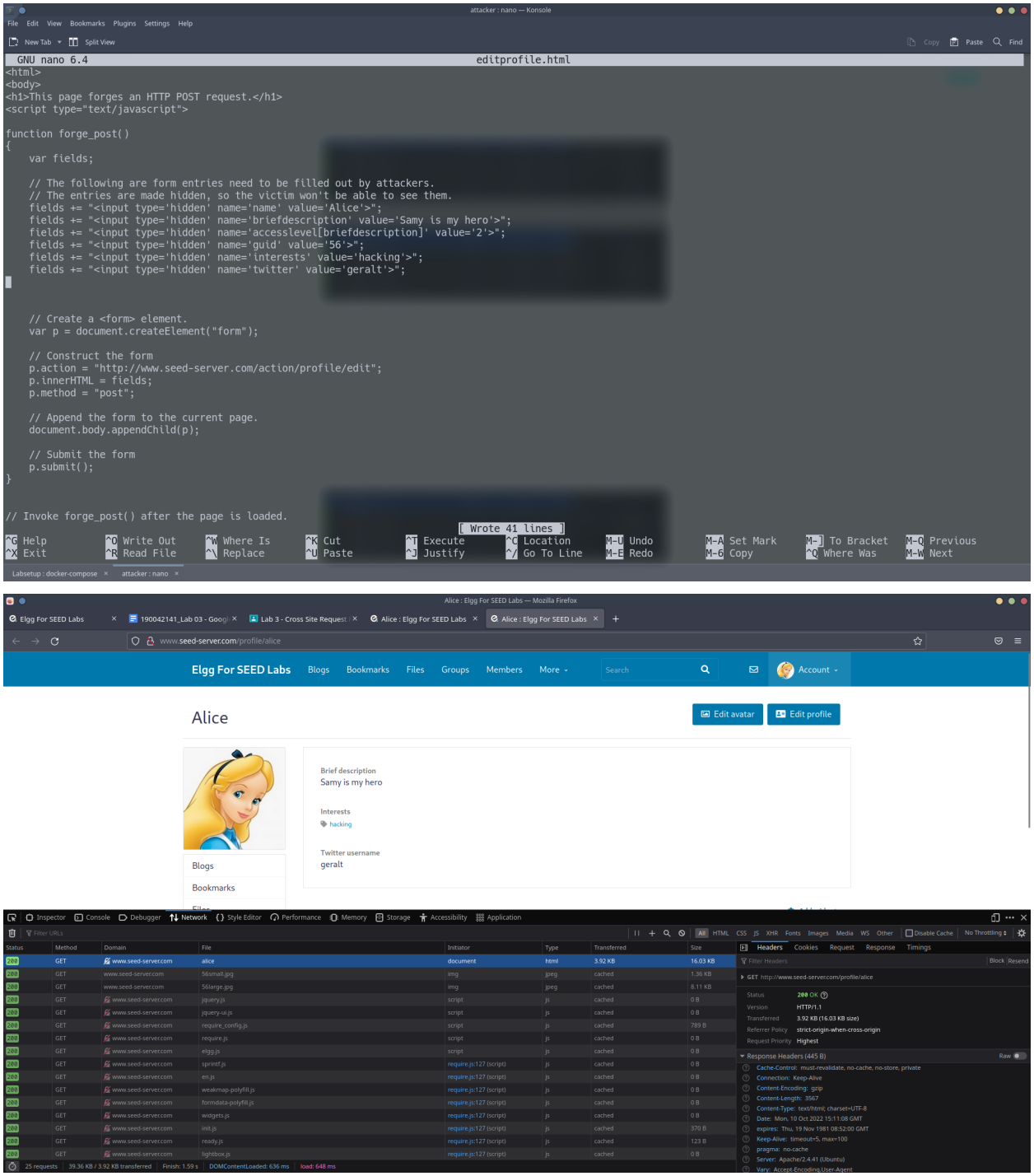








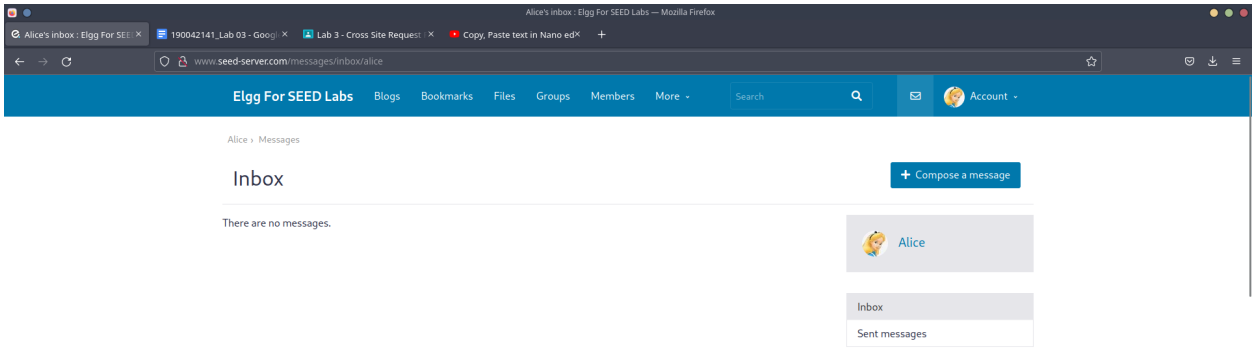
Now after getting the field and its names we modify the previous html code. We add two extra fields to change the interest and twitter field. We give the values with which we want to modify the required profile of Alice

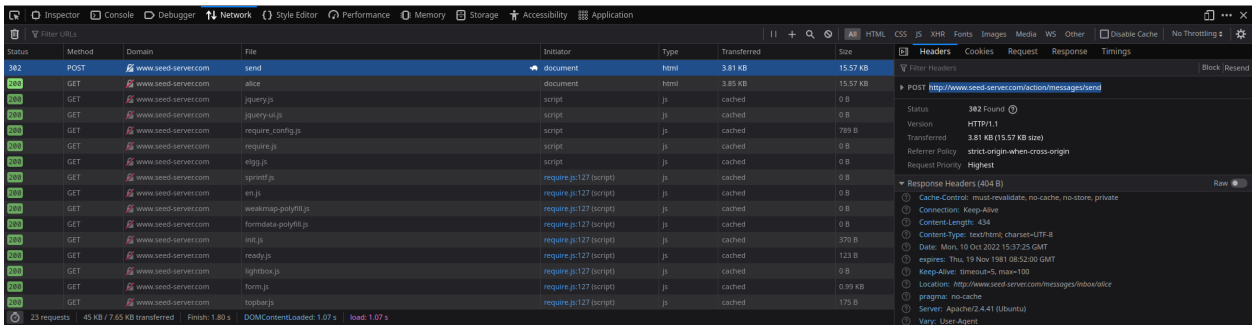


After modifying the malicious html script we send it to Alice. When Alice clicks the malicious link the html code will produce a forged request to modify the profile of Alice

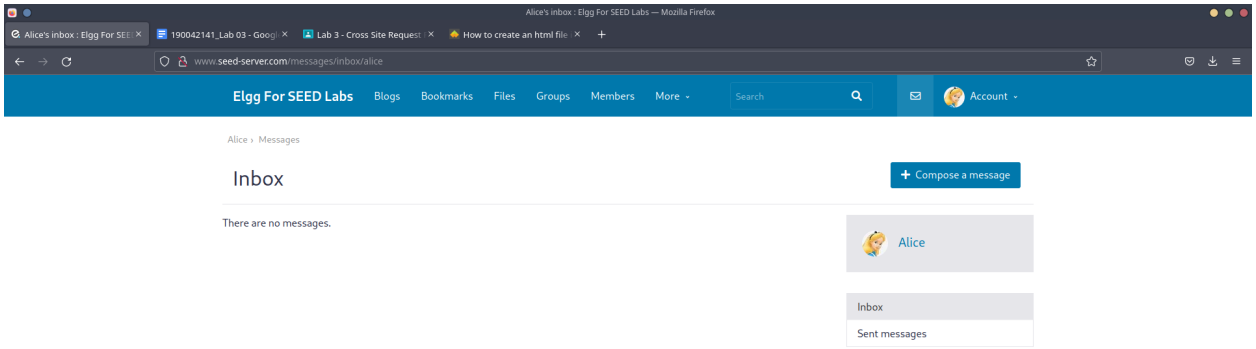
Task-4

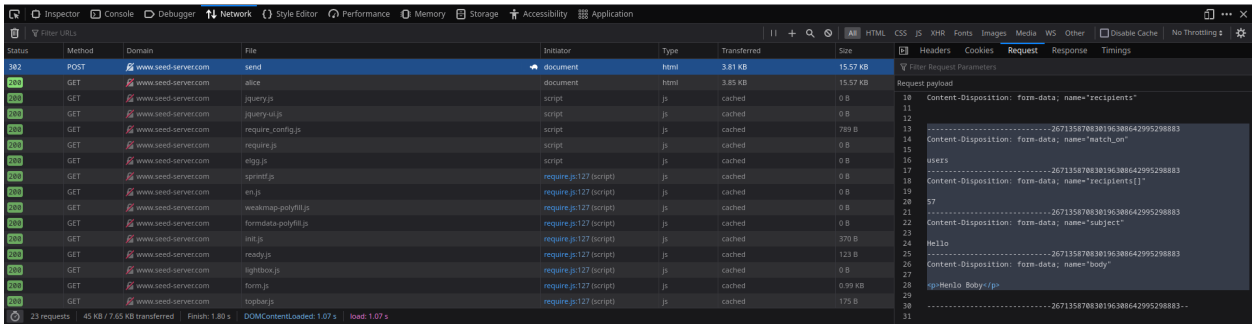
At first we send a dummy message to Boby to check the URL request sent during messaging. As a result we get the required URL- **"http://www.seed-server.com/action/messages/send"**



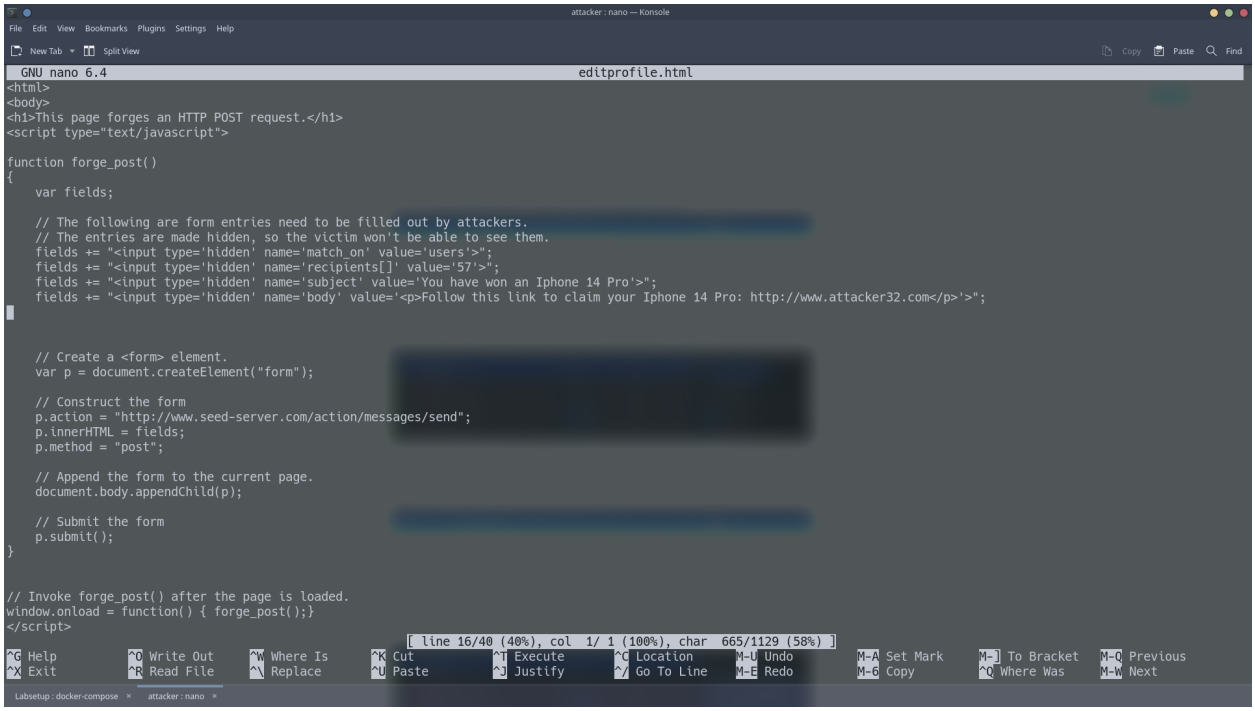


Later we check the fields which are modified during sending the request. As a result we get that the fields match_on,recipients,subject and body

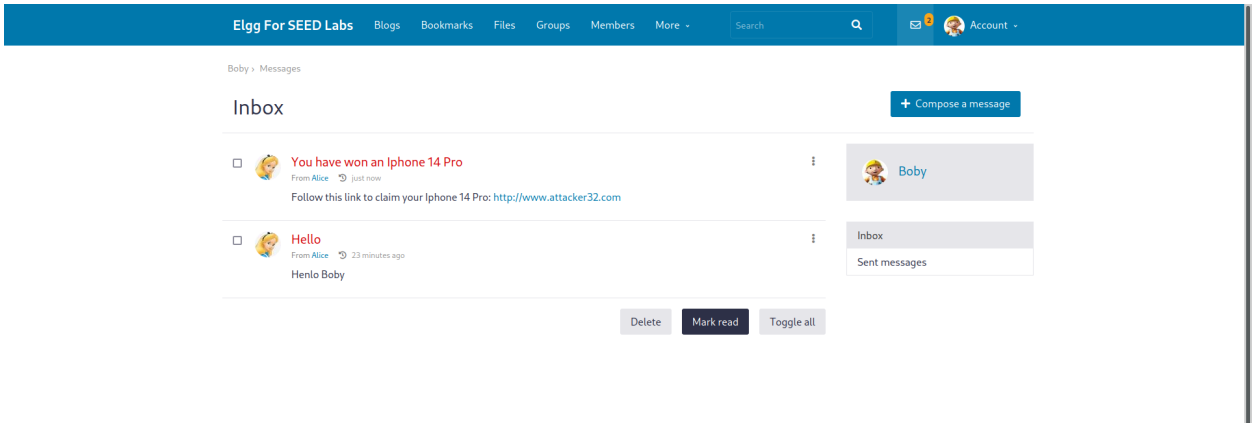




So we change the edit-profile html page with the required fields and the required values against those fields to make a forged request using this malicious html page



Here when Alice goes to that malicious website a message is sent to Boby and the message contains the link **www.attacker32.com**



Task-1

```
CREATE TABLE LOCATION (
LocationCode char(3),
LocationName varchar(50),
CONSTRAINT LOCATION_PK PRIMARY KEY(LocationCode));

CREATE TABLE ROUTE (
RouteId char(3),
RouteDesc varchar(200),
SourceLocationCode char(3),
DesLocationCode char(3),
CONSTRAINT Route_PK PRIMARY KEY(RouteID),
CONSTRAINT Route_FK1 FOREIGN KEY(SourceLocationCode) REFERENCES
LOCATION(LocationCode),
CONSTRAINT Route_FK2 FOREIGN KEY(DesLocationCode) REFERENCES
LOCATION(LocationCode));

create table MULTIPLESTOPOVERS(
RouteID char(3),
Locationcode char(3),
CONSTRAINT MULTIPLESTOPOVERS_PK PRIMARY KEY(RouteID,LocationCode),
CONSTRAINT MULTIPLESTOPOVERS_FK1 FOREIGN KEY(LocationCode) REFERENCES
LOCATION(LocationCode),
CONSTRAINT MULTIPLESTOPOVERS_FK2 FOREIGN KEY(RouteID) REFERENCES
ROUTE(RouteID));

create table CRUISE (
Cruiseid char(3),
cruiseName varchar(30),
NumOfDays varchar(3),
RouteID char(3),
primary key (cruiseid),
CONSTRAINT CRUISE_FK FOREIGN KEY(RouteID) REFERENCES Route(RouteID));

create table SCHEDULE (
ScheduleId char(4),
startdate date,
MaxCapacity varchar(3),
cruiseid char(3),
primary key(scheduleid),
CONSTRAINT SCHEDULE_FK FOREIGN KEY(cruiseid) REFERENCES CRUISE(Cruiseid));
```

```
create table TOUR(
cruiseid char(3),
tourcode char(3),
tourname varchar(25),
tourcost Number(8,2),
tourtype varchar(20),
primary key(cruiseid,tourcode),
CONSTRAINT TOUR_FK FOREIGN KEY(cruiseid) REFERENCES CRUISE(Cruiseid));


create table STAFF (
staffid char(3),
staffname varchar(20),
managerid char(3),
position varchar(20),
StaffPay Number(8,2),
primary key(staffid)
-- CONSTRAINT STAFF_FK FOREIGN KEY(managerid) REFERENCES
STAFF(LocationCode),
);



create table ROSTER (
ScheduleId char(4),
StaffId char(3),
StartDateTime date,
EndDateTime date,
primary key (ScheduleID, Staffid),
CONSTRAINT Roster_FK1 FOREIGN KEY(ScheduleId) REFERENCES
SCHEDULE(ScheduleId),
CONSTRAINT Roster_FK2 FOREIGN KEY(StaffId) REFERENCES STAFF(StaffId)
);




create table QUALIFICATIONS (
```

```
staffid char(3),
skillname varchar(20),
primary key(staffid, skillname),
CONSTRAINT Qualifications_FK FOREIGN KEY(StaffId) REFERENCES STAFF(StaffId)
);


create table MODEL (
modelid char(3),
modelname varchar(15),
modelcapacity varchar(5),
primary key(modelid));


create table VESSEL (
vesselid char(3),
vesselname varchar(20),
vpurchaseyear varchar(5),
modelid char(3),
primary key(vesselid),
CONSTRAINT VESSEL_FK FOREIGN KEY(modelid) REFERENCES MODEL(modelid));

create table BOOKING (
vesselid char(3),
cruiseid char(3),
fromdate date,
todate date,
primary key(vesselid, cruiseid),
CONSTRAINT BOOKING_FK1 FOREIGN KEY(vesselid) REFERENCES VESSEL(vesselid),
CONSTRAINT BOOKING_FK2 FOREIGN KEY(cruiseid) REFERENCES CRUISE(cruiseid)
);

create table SERVICEDOCK(
servicedockid char(4),
servicedockname varchar(30),
primary key(servicedockid));


Create Table SERVICE(
VesselID varchar(4),
ServiceDockID varchar(4),
ServiceDate date,
Primary Key (VesselID, ServiceDockID),
```

```
Foreign key (VesselID) References VESSEL(VesselID),
Foreign key (ServiceDockID) References SERVICEDOCK(ServiceDockID)
);
```

```
insert into LOCATION values('L01','Australia');
insert into LOCATION values('L02','America');
insert into LOCATION values('L03','Switzerland');
insert into LOCATION values('L04','India');
insert into LOCATION values('L05','Sri Lanka');
insert into LOCATION values('L06','Taiwan');
insert into LOCATION values('L07','Japan');
insert into LOCATION values('L08','Egypt');
insert into LOCATION values('L09','Pakistan');
insert into LOCATION values('L10','Maldives');
insert into LOCATION values('L11','Croatia');

insert into ROUTE values('R01','The cruise is 11 days and 10 nights, from Hobart to
Hobart','L01','L01');
insert into ROUTE values('R02','The cruise is 10 days and 9 nights, from Pacific Guatemala to
Panama Caribbean coast','L02','L02');
insert into ROUTE values('R03','The cruise is 15 days and 14 nights, from St Moritz to
Zermatt','L03','L03');
insert into ROUTE values('R04','The cruise is 11 days and 10 nights, from Delhi to
Kerala','L04','L04');
insert into ROUTE values('R05','The cruise is 12 days and 11nights, from Colombo to
Hambantota','L05','L05');
insert into ROUTE values('R06','The cruise is 20 days and 19 nights, from Kaohsiung to
Keelung','L06','L06');
insert into ROUTE values('R07','The cruise is 16 days and 15 nights, from Naze to
Tokyo','L07','L07');
insert into ROUTE values('R08','The cruise is 17 days and 16 nights, from Edfu to Kom
Ombo','L08','L08');
insert into ROUTE values('R09','The cruise is 18 days and 17 nights, from Kemari to
Gawadar','L09','L09');
insert into ROUTE values('R10','The cruise is 18 days and 17 nights, from Baros to
Joali','L10','L10');
```

```
insert into ROUTE values('R11','The cruise is 18 days and 17 nights, from Rab to
Istria','L11','L11');


insert into MULTIPLESTOPOVERS values('R01','L01');
insert into MULTIPLESTOPOVERS values('R02','L02');
insert into MULTIPLESTOPOVERS values('R03','L03');
insert into MULTIPLESTOPOVERS values('R04','L04');
insert into MULTIPLESTOPOVERS values('R05','L05');
insert into MULTIPLESTOPOVERS values('R06','L06');
insert into MULTIPLESTOPOVERS values('R07','L07');
insert into MULTIPLESTOPOVERS values('R08','L08');
insert into MULTIPLESTOPOVERS values('R09','L09');
insert into MULTIPLESTOPOVERS values('R10','L10');
insert into MULTIPLESTOPOVERS values('R11','L11');


insert into CRUISE values('CO1','Hiraeth Cruise','11','R01');
insert into CRUISE values('CO2','Princess Cruise','10','R02');
insert into CRUISE values('CO3','Carnival Liberty','15','R03');
insert into CRUISE values('CO4','Carnival Breeze','11','R04');
insert into CRUISE values('CO5','Coral Princess','12','R05');
insert into CRUISE values('CO6','Welcome Aboard','20','R06');
insert into CRUISE values('CO7','P&O Cruises','16','R07');
insert into CRUISE values('CO8','Skyscraper Cruises','21','R08');
insert into CRUISE values('CO9','Luxury Cruise','19','R09');
insert into CRUISE values('C10','Staycation Cruise','18','R10');
insert into CRUISE values('C11','OverTheWorld Cruise','23','R11');

insert into SCHEDULE values('SC01','2022-10-21','100','CO1');
insert into SCHEDULE values('SC02','2022-11-10','150','CO2');
insert into SCHEDULE values('SC03','2022-11-12','125','CO3');
insert into SCHEDULE values('SC04','2022-11-15','110','CO4');
insert into SCHEDULE values('SC05','2022-11-27','175','CO5');
insert into SCHEDULE values('SC06','2022-11-28','200','CO6');
insert into SCHEDULE values('SC07','2022-11-30','75','CO7');
insert into SCHEDULE values('SC08','2022-12-15','190','CO8');
insert into SCHEDULE values('SC09','2022-12-30','250','CO9');
insert into SCHEDULE values('SC10',NULL,'170','C10');
insert into SCHEDULE values('SC11',null,'185', 'C11');

insert into TOUR values('CO1','TO1','Hireath Business','300.00','Deluxe');
insert into TOUR values('CO2','TO2','Princess Co Business','400.00','Solo');
insert into TOUR values('CO3','TO3','Travel Business','500.00','Family');
```

```
insert into TOUR values('CO4','TO4','Explore Business','200.00','Luxury');
insert into TOUR values('CO5','TO5','Azamara Business','600.00','Adventeruous');
insert into TOUR values('CO6','TO6','Celebrity Business','800.00','General');
insert into TOUR values('CO7','TO7','Cruzify Business','700.00','Sightseeing');
insert into TOUR values('CO8','TO8','SkyFull Business','250.00','Romantic');
insert into TOUR values('CO9','TO9','WCruize Business','200.00','Affordable');
insert into TOUR values('C10','T10','WorldCruize Business','300.00','Buisness');
insert into TOUR values('C11','T11','Sunshine Business','500.00','Buisness');

insert into STAFF values('S01','Shreya Reddy','SO1','Pilot','100');
insert into STAFF values('S02','Candice Lobo','SO2','DJ','200');
insert into STAFF values('S03','Dhavan Mehta','SO3','Host','150');
insert into STAFF values('S04','Shakira Promi','SO4','Instructor','250');
insert into STAFF values('S05','Keval Gandhi','SO5','Waiter','170');
insert into STAFF values('S06','Smriti Kumar','SO6','Assistant','185');
insert into STAFF values('S07','Deepak B','SO7','Housekeeping','225');
insert into STAFF values('S08','Ajay R','SO9','Singer','210');
insert into STAFF values('S09','Arvindh K','S1O','Entertainer','300');
insert into STAFF values('S11','Sandhya Reddy','S11','Crew Member','300');
insert into STAFF values('S12','Madhusudan','S12','Naturalists','300');
insert into STAFF values('S13','Trisha Lokkur','S06','Sport coach','300');
insert into STAFF values('S14','Priyanka P','S07','Guest speaker','300');
insert into STAFF values('S15','Neha Mehta','S13','Clown','300');
insert into STAFF values('S16','Bimal Mehta','S14','Lifeguard','300');


insert into ROSTER values('SC01','S01','2022-10-21 18:30:00','2022-10-24 02:30:00');
insert into ROSTER values('SC02','S02','2022-11-02 18:40:00','2022-11-05 10:10:00');
insert into ROSTER values('SC03','S03','2022-11-05 19:30:00','2022-11-09 12:00:30');
insert into ROSTER values('SC04','S04','2022-11-15 10:20:00','2022-11-18 17:45:00');
insert into ROSTER values('SC05','S05','2022-11-25 11:45:30','2022-11-28 19:00:00');
insert into ROSTER values('SC06','S06','2022-12-10 17:15:00','2022-12-13 22:25:50');
insert into ROSTER values('SC07','S07','2022-12-17 01:00:00','2022-12-20 03:30:00');
insert into ROSTER values('SC08','S08','2022-12-19 02:00:00','2022-12-25 11:30:00');
insert into ROSTER values('SC09','S09','2022-12-25 03:30:35','2022-12-30 04:30:00');
insert into ROSTER values('SC09','S11','2023-01-15 03:30:35','2023-01-30 04:30:00');
insert into ROSTER values('SC10','S12','2023-02-20 03:30:35','2023-02-23 07:30:00');
insert into ROSTER values('SC11','S13','2023-03-11 03:30:45','2023-03-15 10:30:00');
insert into ROSTER values('SC12','S14','2023-04-25 15:30:35','2023-04-30 04:30:00');

insert into QUALIFICATIONS values('S01','Swimming');
insert into QUALIFICATIONS values('S02','Writing');
insert into QUALIFICATIONS values('S03','Speaking');
insert into QUALIFICATIONS values('S04','Hosting');
```

```
insert into QUALIFICATIONS values('S05','Dancing');
insert into QUALIFICATIONS values('S06','Singing');
insert into QUALIFICATIONS values('S07','Cooking');
insert into QUALIFICATIONS values('S08','Managing');
insert into QUALIFICATIONS values('S09','Entertaining');
insert into QUALIFICATIONS values('S01','Illusionist');
insert into QUALIFICATIONS values('S01','Chef');
insert into QUALIFICATIONS values('S02','Doctor');


insert into MODEL values('M01','MODEL-100','150');
insert into MODEL values('M02','MODEL-102','120');
insert into MODEL values('M03','MODEL-104','130');
insert into MODEL values('M04','MODEL-106','100');
insert into MODEL values('M05','MODEL-108','125');
insert into MODEL values('M06','MODEL-110','175');
insert into MODEL values('M07','MODEL-112','200');
insert into MODEL values('M08','MODEL-200','130');
insert into MODEL values('M09','MODEL-2001','140');
insert into MODEL values('M10','MODEL-2002','120');



insert into VESSEL values('V01','Diamond','2010','M01');
insert into VESSEL values('V02','Best Cruises','2011','M02');
insert into VESSEL values('V03','Carnival','2012','M03');
insert into VESSEL values('V04','Sky','2010','M04');
insert into VESSEL values('V05','Golden','2015','M05');
insert into VESSEL values('V06','Rider','2007','M06');
insert into VESSEL values('V07','Mignon','2016','M07');
insert into VESSEL values('V08','Happy','2017','M08');
insert into VESSEL values('V09','Voyages','2019','M09');
insert into VESSEL values('V10','LovelyCruise','2021','M10');




insert into BOOKING values('V01','CO1','2022-10-22','2022-11-22');
insert into BOOKING values('V02','CO2','2022-11-20','2022-12-20');
insert into BOOKING values('V03','CO3','2023-01-15','2023-02-20');
insert into BOOKING values('V04','CO4','2023-02-17','2023-03-17');
insert into BOOKING values('V05','CO5','2023-03-5','2023-04-30');
insert into BOOKING values('V06','CO6','2023-04-19','2023-05-22');
```

```
insert into BOOKING values('V07','CO7','2023-05-07','2023-06-10');
insert into BOOKING values('V08','CO9','2023-06-08','2023-07-8');
insert into BOOKING values('V09','CO9','2023-07-17','2023-08-21');
```

```
insert into SERVICEDOCK values('SD01','Dock Mac');
insert into SERVICEDOCK values('SD02','Dock Carnival');
insert into SERVICEDOCK values('SD03','Dock Princess');
insert into SERVICEDOCK values('SD04','Dock Diamond');
insert into SERVICEDOCK values('SD05','Dock BestCruise');
insert into SERVICEDOCK values('SD06','Dock Skies');
insert into SERVICEDOCK values('SD07','Dock TravelSeas');
insert into SERVICEDOCK values('SD08','Dock Happiness');
insert into SERVICEDOCK values('SD09','Dock AZCruize');
```

```
insert into SERVICE values('V01','SD01','2022-10-23');
insert into SERVICE values('V02','SD02','2022-11-24');
insert into SERVICE values('V03','SD03','2022-11-15');
insert into SERVICE values('V04','SD04','2022-12-04');
insert into SERVICE values('V05','SD05','2023-01-11');
insert into SERVICE values('V06','SD06','2023-02-18');
insert into SERVICE values('V07','SD07','2023-03-31');
insert into SERVICE values('V08','SD08','2023-04-14');
insert into SERVICE values('V09','SD09','2023-04-21');
```

Task-2

```
select TourName, TourType, concat('$ ',tourcost) as TourCost FROM tour
order by tourcost desc;
```

Task-3

```
select VesselId,FromDate,ToDate, datediff(day, todate, fromdate) AS date_difference from
booking;
```

Task-4

```
select vesselid, vesselname, vessel.vpurchaseyear, vessel.modelid
from vessel, model
where vessel.modelid=model.modelid
and model.modelname like 'Model-200%'
and vessel.vpurchaseyear > '2015';
```

Task-5

```
select distinct vessel.vesselname
from vessel join booking
using (vesselid)
where datediff(todate, fromdate) > 10
and vpurchaseyear between 2015 and 2020;
```

Task-6

```
select modelname, modelcapacity
from model
where modelid in (select modelid from vessel where vpurchaseyear>'2018');
```

Task-7

```
select distinct s1.staffname, s2.staffname
from staff s1, staff s2, roster
where s1.staffid=s2.managerid
and roster.enddatetime>'1900';
```

Task-8

```
select modelcapacity as number_of_cruises, vesselname as vessel_name
from vessel v,model m
where v.modelid=m.modelid
order by modelcapacity desc;
```

Task-9

```
select cruise.cruiseid, cruisename, numofdays, routeid
from cruise, schedule
where cruise.cruiseid=schedule.cruiseid
and schedule.startdate is null;
```

Task-10

```
Select CruiseName, t.TourCode, TourName, Count(c.CruiseId) as TotalAmountOfCruise,
TourCost, c.NumOfDays*t.TourCost*1.1 as TotalTourCostWithTax
```

```
from CRUISE c join
TOUR t
On c.CruiseID=t.CruiseId
Where c.NumOfDays*t.TourCost*1.1>3000
group by t.TourCode;
```

Task-11

```
select s.StaffID, s.StaffName, count(SkillName) as NumberOfSkills
from QUALIFICATIONS q
join staff s
on s.StaffID = q.StaffID
Group by StaffID
Having count(SkillName)>=2;
```

Task-12

```
Select s.StaffID, StaffName, datediff(EndDateTime,StartDateTime) as MaximumNumberOfDays,
TourCost, round(TourCost*1.027, 2) as UpdatedTourCost
from Staff s
join ROSTER r
On s.StaffID=r.StaffID
join SCHEDULE sc
On sc.ScheduleID=r.ScheduleID
join CRUISE c
On c.CruiseID=sc.CruiseID
join TOUR t
On t.CruiseID=c.CruiseID
where TourType= 'Business';
```

Task-13

```
select c.CruiseID, TourCode, TourName, TourCost, TourType, a.LocationName as
SourceLocationName, b.LocationName as DesLocationName from TOUR t
join Schedule sc
On t.CruiseID=sc.CruiseID
join CRUISE c
On sc.CruiseID=c.CruiseID
Join ROUTE r
On r.RouteID=c.RouteID
Join LOCATION a, LOCATION b
Where (r.SourceLocationCode=a.LocationCode
and r.DesLocationCode=b.LocationCode)
```

```
And TourCost>250
And (month(StartDate)=01
or year(StartDate)=2020 or year(StartDate)=2022);
```

Task-14

Write a query to print the names of the service dock that have had a cruise with more than one source location.*/

```
select ServiceDockName
from servicedock sd
join service s
On s.ServiceDockID=sd.ServiceDockID
join BOOKING b
On b.VesselID=s.VesselID
join CRUISE c
On c.CruiseID=b.CruiseID
join ROUTE r
On r.RouteID=c.RouteID
Having count(SourceLocationCode)>1;    -- servicedock,service, booking, cruise, sourcelocation(route) problems arising, ust 1 result

select ServiceDockID, s.VesselId
from SERVICE s
JOIN booking b
on b.VesselID=s.VesselID;
```

Task-15

```
Select c.CruiseID, CruiseName, NumOfDays, TourName
from CRUISE c
Inner Join TOUR t
on c.CruiseID=t.CruiseID
Where NumOfDays>10
And TourCode is not Null
Group by CruiseID;
```

Task-16

```
Select round(avg(TourCost),0) as AverageCostOfEarnings
from TOUR t
```

```
join SCHEDULE sc
On sc.CruiseID=t.CruiseID
join ROSTER r
On r.ScheduleID=sc.ScheduleID
Where month(r.EndDateTime)>'09'
and year(r.EndDateTime)='2022';
```

Task-17

```
Select s.StaffName, m.StaffName as ManagerName, CruiseName
from STAFF s
join STAFF m
On s.ManagerID=m.StaffID
join ROSTER r
On s.StaffID=r.StaffID
join SCHEDULE sc
on sc.SCHEDULEID=r.ScheduleID
join CRUISE c
on sc.CruiseID=c.CruiseID
Where time(StartDateTime) between '12:00:00' and '15:00:00'
and time(EndDateTime) between '06:00:00' and '12:00:00';
```

Task-18

```
Select c.CruiseID, CruiseName, SourceLocationCode, a.locationName as
SourceLocationName, DesLocationCode, b.locationName as DesLocationName
from Location a
join ROUTE r
On a.LocationCode=r.SourceLocationCode
join LOCATION b
On b.LocationCode=r.DesLocationCode
Join CRUISE c
On c.RouteID=r.RouteID
Where a.LocationName= 'Sydney'
or b.LocationName= 'Sydney';
```

Task-19

```
Select year(EndDateTime) as Year, StaffPay as AvgSalaryOfStaffPerYear, StaffName
from STAFF s
join ROSTER r
On s.StaffID=r.StaffID
Group by StaffName
```

```
Having StaffPay<(select avg(StaffPay)
                from STAFF);
```

Task-20

```
Select count(c.CruiseID) as CruisePerServiceDock, ServiceDockName, count(st.StaffID) as
StaffPerServiceDock
from SERVICEDOCK sd
JOIN SERVICE s
On sd.ServiceDockID=s.ServiceDockID
 join VESSEL v
On s.VesselID=v.VesselID
 join BOOKING b
On b.VesselID=v.VesselID
join CRUISE c
On b.CRUISEID=c.CRUISEID
 join SCHEDULE sc
On sc.CRUISEID=c.CRUISEID
join ROSTER r
On r.ScheduleID=sc.ScheduleID
Join STAFF st
On st.StaffID=r.StaffID
Where st.StaffName not like '% E%'
Group by ServiceDockName;
```